Kwanty

Jacek Markiewicz

2025/2026

Imagination will often carry us to worlds that never were. But without it we go nowhere.

Carl Sagan

Spis treści

1 Wstęp

Definicja

Macierz U jest **unitarna** gdy $UU^\dagger = U^\dagger U = I$.
Macierz U nazywamy **hermitowską** jeśli $U = U^\dagger$.

Macierze unitarne są dla nas istotne o tyle, że bramki kwantowe są właśnie operatorami unitarnymi.

Lemat

Macierze unitarne zachowują normę L_2.

Dowód. Niech v to dowolny element taki, że $\|v\|_2 = 1$. Poza tym niech U - macierz unitarna, czyli $UU^\dagger = U\dagger U = I$. Wtedy:

$$\begin{aligned}\|v\|_2 &= \langle v|v\rangle = \langle v| \cdot |v\rangle = \langle v| \cdot I \cdot I \cdot |v\rangle = \langle v| \cdot (U^\dagger U) \cdot (U^\dagger U) \cdot |v\rangle \\ &= \Big(\langle v| U^\dagger\Big)\Big(U|v\rangle\Big) = \Big(U|v\rangle\Big)^\dagger\Big(U|v\rangle\Big) = \langle Uv|Uv\rangle = \|Uv\|_2\end{aligned}$$

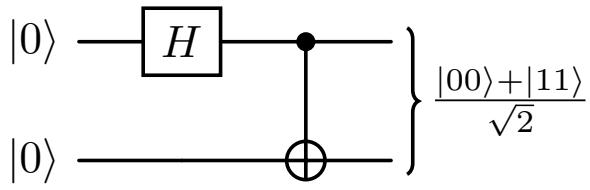
□

1.1 Stany Bella

Definicja

Stanami Bella określamy 2-kubitowe stany $\frac{|00\rangle \pm |11\rangle}{\sqrt{2}}$ oraz $\frac{|01\rangle \pm |10\rangle}{\sqrt{2}}$.

Stany Bella są najprostszym przykładem splątania kwantowego. Zmierzenie wartości pierwszego kubitu sprawi, że wartość drugiego kubitu zostanie już ustalona. Jednak gdybyśmy nie zmierzyli pierwszego kubitu, to pomiar drugiego z nich mógłby wyprodukować zarówno 0 jak i 1.
Nietrudno stworzyć taki stan. Czyni to na przykład następujący układ:



(1)

Jest tak, bo:

$$|00\rangle \xmapsto{H\otimes I} \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)|0\rangle = \frac{|00\rangle + |10\rangle}{\sqrt{2}} \xmapsto{CX} \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

2 Algorytm Deutscha-Jozsy

Problem

Dana jest czarna skrzynka w postaci funkcji $f : \{0,1\}^n \to \{0,1\}$, o której wiemy, że zachodzi jedno z poniższych:

- f jest funkcją *stałą* (zawsze zwraca 0 lub zawsze zwraca 1) lub
- f jest funkcją *zbalansowaną* (przypisuje 0 takiej samej ilości elementów co wartość 1)

Naszym zadaniem jest stwierdzenie, czy f jest stała czy zbalansowana.
Chcemy dodatkowo dokonać tego wykonując możliwie mało ewaluacji funkcji f.

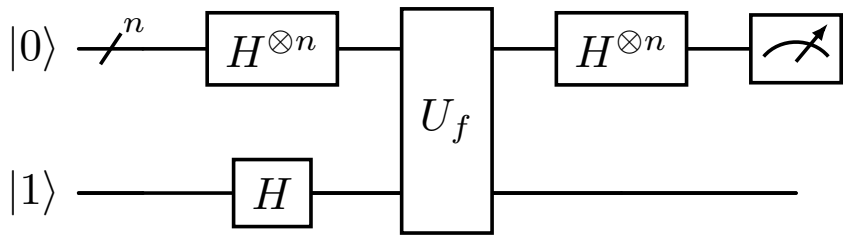
Klasyczne podejście

Na komputerze klasycznym konieczne jest $\mathcal{O}(2^n)$ operacji. Wynika to z tego, że pesymistycznie pierwsze 2^{n-1} sprawdzeń wartości może zakończyć się tym samym wynikiem, niezależnie jakiego rodzaju jest to funkcja.

Innym pomysłem jest algorytm randomizowany, tj. wylosowanie k wartości i ich zewaluowanie. Jeśli otrzymaliśmy zarówno 0 jak i 1 jako wyniki, to funkcja na pewno była zbalansowana. Wpp. strzelamy, że była funkcją stałą. Jest to niezła metoda, gdyż prawdopodobieństwo pomyłki (czyli uznania f za stałą gdy była tylko zbalansowaną) wynosi 2^{-k}. Wymagało to dodatkowo tylko k ewaluacji funkcji f.

Kwantowe podejście

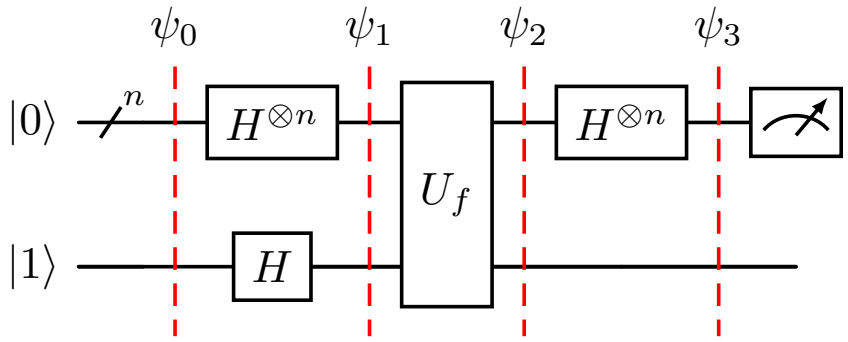
My jednak będziemy jeszcze lepsi i stworzymy algorytm deteministyczny, który da nam zawsze poprawną odpowiedź wykonując tylko jedną ewaluację funkcji f. Bez owijania w bawełnę oto nasz szukany obwód kwantowy:



(2)

Jeśli zmierzymy $|0\rangle^{\otimes n}$ to funkcja była stała. W przeciwnym przypadku była zbalansowana.

Dowód.
Żeby zobaczyć dlaczego ten układ rozwiązuje nasz problem musimy przeanalizować co się dzieje w każdym jego kroku:



(3)

- $|\psi_0\rangle = |0^{\otimes n}\rangle\,|1\rangle$
- $|\psi_1\rangle$:
  Przypomnijmy sobie jak działa bramka Hadamarda:

$$H\,|0\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle$$

$$H\,|1\rangle = \frac{1}{\sqrt{2}}\,(|0\rangle - |1\rangle)$$

Podstawiając

$$|\psi_1\rangle = H|\psi_0\rangle = (H|0\rangle^{\otimes n})(H|1\rangle) = \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|0\rangle - |1\rangle)$$

- $|\psi_2\rangle$:
  Wyrocznia U_f zachowuje się tak:

$$U_f|x\rangle_n|y\rangle = |x\rangle_n|y \oplus f(x)\rangle$$

$$\Rightarrow |\psi_2\rangle = U_f|\psi_1\rangle = \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|f(x)\rangle - |1 \oplus f(x)\rangle)$$

  ale $f(x)$ przyjmuje tylko wartości 0 lub 1, zatem możemy zapisać to sprytniej

$$= \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle(|0\rangle - |1\rangle)$$

- $|\psi_3\rangle$:
  Bramka Hadamarda potrafi jeszcze tak:

$$H|x\rangle = \frac{1}{\sqrt{2^n}}\sum_{i=0}^{2^n-1}(-1)^{x\cdot i}|i\rangle$$

  gdzie $x \cdot i$ oznacza iloczyn skalarny, czyli $x \cdot i = x_1 i_1 \oplus ... \oplus x_n i_n$.
  Ostatni wynikowy kubit nie jest nam potrzebny, więc go ignorujemy. Zatem niech ψ_2' oznacza stan ψ_2 z uciętym ostatnim bitem, analogicznie ψ_3'. Wtedy:

$$|\psi_2'\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle$$

  Zatem ostatnia transformacja daje nam

$$|\psi_3'\rangle = H|\psi_2'\rangle = \frac{1}{2^n}\sum_{x=0}^{2^{n-1}}(-1)^{f(x)}\cdot\sum_{i=0}^{2^n-1}(-1)^{x\cdot i}|i\rangle$$

$$= \frac{1}{2^n}\sum_{i=0}^{2^n-1}\left[\sum_{x=0}^{2^n-1}(-1)^{f(x)}(-1)^{x\cdot i}\right]|i\rangle$$

  Ale teraz zauważmy, że prawdopodobieństwo zmierzenia stanu $0^{\otimes n}$ wynosi

$$\left|\frac{1}{2^n}\sum_{x=0}^{2^n-1}(-1)^{f(x)}\right|^2$$

  Wystarczy teraz przeanalizować przypadki:
  1. Jeśli f jest stała, to $\sum_{x=0}^{2^n-1} = \pm 2^n$, czyli prawdopodobieństwo wyjdzie równe 1.
  2. Jeśli f jest zbalansowana, to $\sum_{x=0}^{2^n-1} = 0$, bo wyrazów sumy równych 1 jest tyle samo co równych -1. Zatem prawdopodobieństwo wyniesie 0.

  Czyli by stwierdzić czy f jest stała wystarczy sprawdzić czy na końcu zmierzymy $0^{\otimes n}$. Jeśli zmierzymy cokolwiek innego to znaczy, że była zbalansowana.

□

2.1 Bernstein–Vazirani

Problem

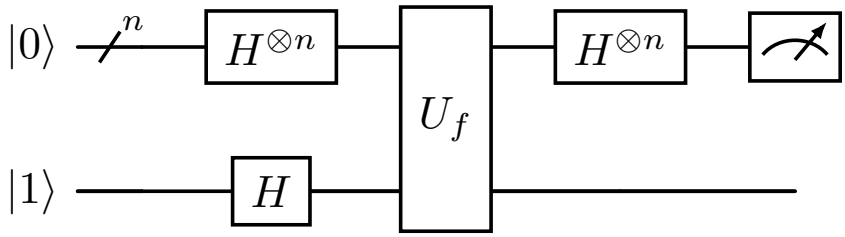
Dana jest czarna skrzynka w postaci funkcji $f : \{0,1\}^n \to \{0,1\}$. Mamy zapewnione, że f jest postaci

$$f(x) = x \cdot s$$

gdzie $\cdot$ jest iloczynem skalarnym *mod* 2, a s - nieznanym wektorem z $\{0,1\}^n$.
Zadanie polega na wyznaczeniu s.
Dodatkowo chcemy tego dokonać wykonując możliwie mało zapytań o funkcję f.

Łatwo pokazać, że klasyczne podejście zawsze wymaga co najmniej n zapytań o wartość $f(x)$. Wynika to z faktu, że przestrzeń potencjalnych wartości s jest wielkości 2^n, a każde zapytanie się o wartość $f(x)$ odrzuca pewną część tej przestrzeni, pesymistycznie co najwyżej połowę. Czyli aby być pewnym wartości s trzeba wykonać co najmniej $\lg(2^n) = n$ zapytań.

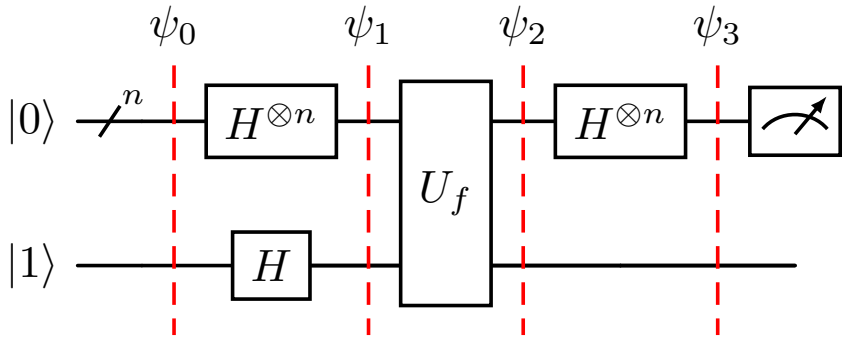
My będziemy lepsi i wyznaczymy deterministycznie wartość s używając tylko jednej ewaluacji funkcji f.
By to osiągnąć użyjemy następującego układu:



(4)

Po jego uruchomieniu szukany przez nas wektor s magicznie pojawi się na mierzonym wyjściu.

Dowód. By to pokazać, musimy przeanalizować działanie układu.



(5)

A zatem po kolei:

- $|\psi_0\rangle = |0\rangle_n |1\rangle$
- Aplikujemy Hadamarda. Pamiętamy, że:

  $$H|0\rangle_n = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle, \qquad H|1\rangle = |-\rangle$$

  Zatem

  $$|\psi_1\rangle = H|\psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle |-\rangle$$

- Aplikujemy wyrocznię U_f. Wiemy, że działa ona tak:

  $$U_f |x\rangle_n |y\rangle = |x\rangle_n |y \otimes f(x)\rangle$$

  Czyli:

  $$U_f |x\rangle_n |-\rangle = U_f |x\rangle_n \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) = |x\rangle_n \left(\frac{|f(x)\rangle - |f(x)\oplus 1\rangle}{\sqrt{2}}\right)$$

  Zauważmy, po "zignorowaniu" ostatniego kubitu jedynym widocznym wynikiem $f(x)$ w układzie jest zmieniona faza (+1 lub -1).
  Niech $|g\rangle$ oznacza stan ostatniego bitu. Wtedy mamy:

  $$|\psi_2\rangle = U_f |\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{f(x)} |x\rangle |g\rangle$$

- Ponownie Hadamard. Pamiętając, że $H|k\rangle = \frac{1}{\sqrt{2^n}} \sum_{i=0}^{2^n-1} (-1)^{k\cdot i} |i\rangle$ otrzymujemy:

  $$|\psi_3\rangle = H|\psi_2\rangle = \frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{y=0}^{2^n-1} (-1)^{f(x)+x\cdot y} |y\rangle |g\rangle$$

Dla uproszczenia analizy zapomnijmy o kubicie $|g\rangle$. Zauważmy, że dla konkretnego y:

$$\frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)+x\cdot y} = \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{x\cdot s + x\cdot y} = \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{x\cdot(s\oplus y)}$$

Łatwo zauważyć, że gdy $s = y$ to $s \oplus y = 0$ i wyrażenie jest równe 1. Jeśli natomiast $s \neq y$ to $s \oplus y$ będzie przyjmowało wartości 0 i 1 dla tej samej ilości wyrazów, więc wyrażenie się wyzeruje.
Czyli prawdopodobieństwo zmierzenia stanu $|s\rangle$ na pierwszych n kubitach wynosi 1. □

3 Algorytm Grovera

Problem

Dana czarna skrzynka w postaci funkcji $f : \{0,1\}^n \to \{0,1\}$. Funkcja ta zawsze przyjmuje wartość 0, poza jednym nieznanym argumentem ω, dla którego $f(\omega) = 1$.
Zadanie polega na znalezieniu wartości ω.

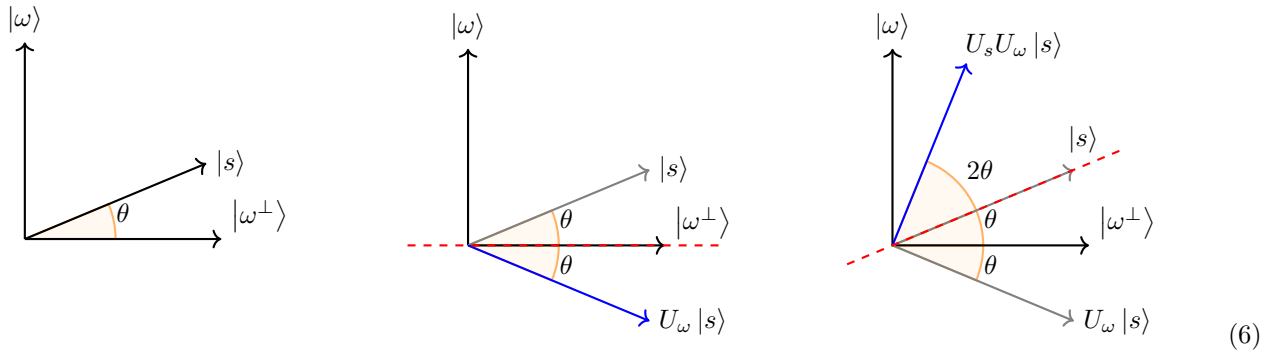
W przypadku klasycznym widać, że potrzebne jest średnio $\frac{N}{2}$, a pesymistycznie $N-1$ ewaluacji funkcji f do wyznaczenia ω.
My dokonamy czegoś niesamowitego i wyznaczymy nieznaną wartość używając jedynie $\mathcal{O}(\sqrt{N})$ ewaluacji f.
Użyjemy do tego dwóch operatorów:

- $U_\omega := I - 2\left|\omega\right\rangle\!\left\langle\omega\right|$
- $U_s := 2\left|s\right\rangle\!\left\langle s\right| - I$

Oba są odbiciami Householdera. Oznacza to tyle, że jeśli rozważymy płaszczyznę rozpiętą przez wektory $|\omega\rangle$ i $|\omega^\perp\rangle$ to:

- U_ω odbija każdy wektor względem płaszczyzny $|\omega^\perp\rangle$ (prostopadłej do $|\omega\rangle$).
- U_s odbija wektor względem $|s\rangle$

Załóżmy, że początkowo dany jest wektor $|s\rangle$, którego początkowy kąt z $|\omega^\perp\rangle$ wynosi θ. Wtedy przekształcenie U_sU_ω graficznie wygląda następująco:



(6)

Przesunęliśmy się zatem w stronę $|\omega\rangle$ o kąt 2θ. Okazuje się, że każde kolejne zaaplikowanie U_sU_ω również przesuwa nas w stronę $|\omega\rangle$ o 2θ. Zatem możemy do skutku aplikować U_sU_ω, a potem zmierzyć końcowy stan, który prawie na pewno będzie naszym szukanym $|\omega\rangle$.
Pozostaje jeszcze wybrać taki początkowy wektor $|s\rangle$, by $\theta > 0$. Z braku lepszych opcji możemy ustalić $|s\rangle = H\left|0\right\rangle_n = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}|k\rangle$. Sprawdźmy ile wtedy wynosi θ:

$$\cos\theta = \frac{\left\langle s|\omega^\perp\right\rangle}{||s|| \cdot ||\omega^\perp||} = \left\langle s|\omega^\perp\right\rangle$$

Wiemy też, że

$$|s\rangle = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}|k\rangle = \frac{1}{\sqrt{N}}|\omega\rangle + \frac{1}{\sqrt{N}}\sum_{k=0,k\neq\omega}^{N-1}|k\rangle = \frac{1}{\sqrt{N}}|\omega\rangle + \frac{\sqrt{N-1}}{\sqrt{N}}|\omega^\perp\rangle$$

Zatem

$$\left\langle s|\omega^\perp\right\rangle = \frac{1}{\sqrt{N}}\left\langle\omega|\omega^\perp\right\rangle + \frac{\sqrt{N-1}}{\sqrt{N}}\left\langle\omega^\perp|\omega^\perp\right\rangle = \frac{\sqrt{N-1}}{\sqrt{N}}$$

$$\Rightarrow \theta = \arccos\sqrt{\frac{N-1}{N}}$$

Z jedynki trygonometrycznej mamy

$$\theta = \arcsin\frac{1}{\sqrt{N}}$$

Ale dla małych kątów $\arcsin x \approx x$, czyli $\theta \approx \frac{1}{\sqrt{N}}$.
Za każdym razem przesuwamy się o kąt 2θ, a chcemy aby kąt z początkowego ≈ 0 stał się równy $\pi/2$. Łatwo zatem wyliczyć ilość potrzebnych zaaplikowań U_sU_ω:

$$\frac{\pi/2}{2\theta} \approx \frac{\pi/2 \cdot \sqrt{N}}{2} = \boxed{\frac{\pi}{4}\sqrt{N}} = \mathcal{O}(\sqrt{N})$$

Mamy już zatem gotowy algorytm (Grovera):

1. Zainicjuj $|s\rangle := H\left|0\right\rangle_n$
2. Zaaplikuj $\frac{\pi}{4}\sqrt{N}$ razy U_sU_ω
3. Zmierz końcowy rejestr

Definicja. (BPP)

Bounded-error polynomial time (**BPP**) to klasa problemów decyzyjnych, które można rozwiązać w czasie wielomianowym, z prawdopodobieństwem pomyłki co najwyżej $\frac{1}{3}$ dla dowolnej z instancji.

Definicja. (BQP)

Bounded-error quantum polynomial time (**BQP**) to klasa problemów decyzyjnych, które można rozwiązać w czasie wielomianowym na komputerze kwantowym, z prawdopodobieństwem pomyłki co najwyżej $\frac{1}{3}$ dla dowolnej z instancji.

Zachodzi następujące:

$$P \subseteq BPP \subseteq BQP \subseteq PSPACE$$

Na dzień dzisiejszy dla żadnego z powyższych zawierań nie wiadomo czy są one ostre czy słabe.

Definicja. (Uniwersalny zbiór bramek)

Zbiór bramek S nazywamy **uniwersalnym** jeśli dla każdej macierzy unitarnej $U \in \mathbb{C}^{N \times N}$ oraz $\varepsilon > 0$ istnieje obwód C złożony wyłącznie z bramek z S, nieużywający dodatkowych kubitów, taki, że dla każdego stanu $|\psi\rangle$ zachodzi

$$||U|\psi\rangle - C|\psi\rangle|| \leq \varepsilon$$

Intuicyjnie zbiór bramek jest uniwersalny, gdy możemy zasymulować z dowolną prezycją dowolną operację U możliwą na komputerze kwantowym.

4 Kubityzacja

Macierze, których używamy w układach kwantowych muszą być unitarne. Co jednak w sytuacji, gdy chcemy wykonać operację niebędącą unitarną? Nie wszystko stracone - wystarczy zakodować wymarzoną przez nas macierz A w większej macierzy U_A, poprzez dorzucenie dodatkowych kubitów (tzw. *ancillas*):

$$U_A = \begin{pmatrix} A & * \\ * & * \end{pmatrix}$$

gdzie wartości $*$ nas nie interesują, bo dotykają *ancilla* kubitów. Nie interesują nas ich konkretne wartości, ale co ważne są one dobrane tak, że macierz całościowo jest unitarna.

Definicja. (Block-encoding)

Mówimy, że U_A to (α, a, ε)-**block encoding** macierzy A, jeśli:

$$||A - \alpha({}_a\langle 0| \otimes \mathbb{I}_s) U_A (|0\rangle_a \otimes \mathbb{I}_s)|| \leq \varepsilon$$

gdzie $\alpha \in \mathbb{R}^+$, $\varepsilon \in \mathbb{R}^+$, $|0\rangle_a$ oznacza $|0\rangle^{\otimes a}$

Zapis $({}_a\langle 0| \otimes \mathbb{I}_s) U_A (|0\rangle_a \otimes \mathbb{I}_s)$ oznacza tyle, że "wycinamy" z U_A fragment reprezentujący macierz A.
Powyższa definicja jest dość ogólna. Zezwalamy by nasza macierz różniła się od wynikowej (w normie spektralnej) o co najwyżej ε. W definicji uwzględniamy również pewne przeskalowanie macierzy A o współczynnik α. Do samego zakodowania macierzy U_A potrzebujemy też a *ancilla*-kubitów.

Przykład. Dla macierzy unitarnej A, samo A jest jej $(1, 0, 0)$-block encoding'iem

Lemat

Niech U_A i U_B to odpowiednio $(\alpha, a, \varepsilon_A)$-BE dla macierzy A i $(\beta, b, \varepsilon_B)$-BE dla macierzy B.
Niech $U_{AB} := (I_b \otimes U_A)(I_a \otimes U_b)$.
Wtedy U_{AB} jest $(\alpha\beta, a+b, \alpha\varepsilon_B + \beta\varepsilon_A + \varepsilon_A\varepsilon_B)$-BE dla macierzy AB.

4.1 LCU

Linear Decomposition of Unitaries (LCU) to metoda wyznaczania block-encoding'u dla zadanej macierzy A. Sam algorytm jest dość nieskomplikowany. Najpierw, zapiszmy A jako sumę macierzy unitarnych U_k:

$$A = \sum_{k=0}^{N-1} \alpha_k U_k$$

gdzie $N = 2^{n+a}$, $\alpha_i \in \mathbb{R}$.
Teraz zdefiniujmy operatory $PREP$ i SEL:

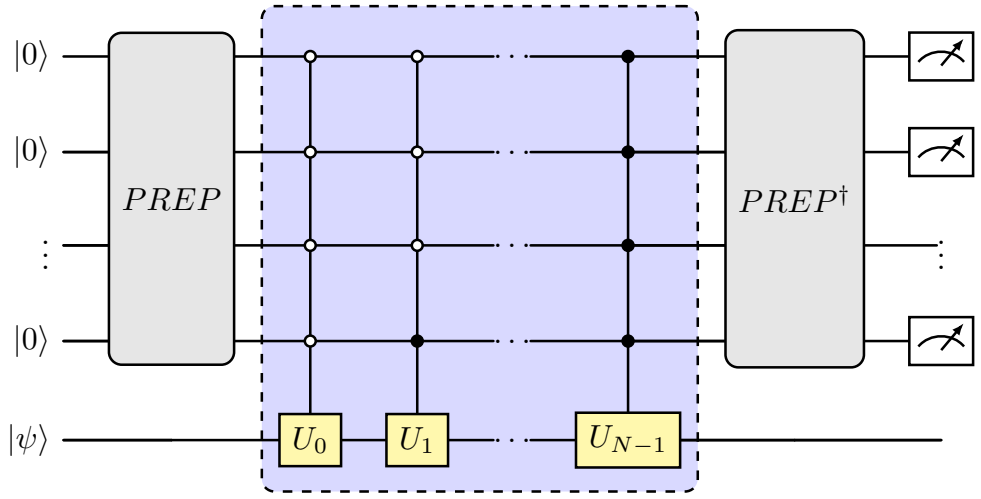
$$PREP\,|0\rangle = \sum_{k=0}^{N-1} \sqrt{\frac{|\alpha_k|}{\lambda}}|\,|k\rangle$$
$$SEL\,|k\rangle\,|\psi\rangle = |k\rangle\,U_k\,|\psi\rangle$$

gdzie współczynnik normalizacji λ jest zdefiniowany jako $\lambda = \sum_{k=0}^{N-1} |\alpha_k|$.

Zdefiniujmy teraz U_A jako

$$U_A := PREP^\dagger \cdot SEL \cdot PREP$$

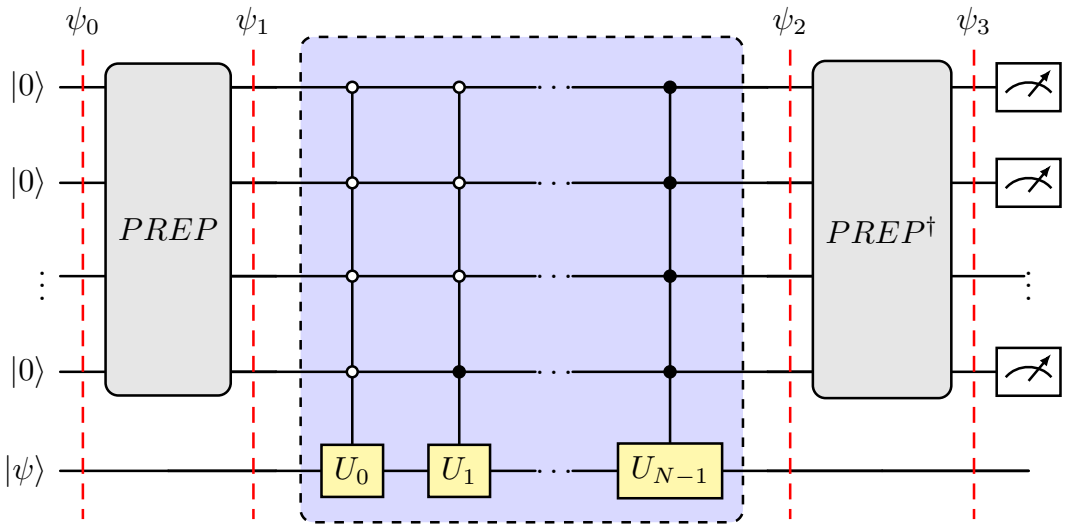
Okazuje się, że U_A jest $(\lambda, a, 0)$-block encoding'iem A. Oznacza to, że możemy skonstruować taki układ:



SELECT (7)

Jeśli teraz go uruchomimy, oraz na wszystkich (górnych) a miernikach zmierzymy $|0\rangle$, to końcowy stan na ostatniej (dolnej) linii będzie równy $\frac{A}{\lambda}|\psi\rangle$.

Dowód. Przeanalizujmy dokładnie co się dzieje w tym układzie



SELECT (8)

- $|\psi_0\rangle = |0\rangle_a |\psi\rangle$
- Aplikujemy $PREP$:

$$|\psi_1\rangle = PREP\,|0\rangle\,|\psi_0\rangle = \sum_{k=0}^{N-1} \sqrt{\frac{|\alpha_k|}{\lambda}}\,|k\rangle\,|\psi\rangle$$

- Aplikujemy SEL:

$$|\psi_2\rangle = SEL\,|\psi_1\rangle = \sum_{k=0}^{N-1} \sqrt{\frac{|\alpha_k|}{\lambda}} SEL\,|k\rangle\,|\psi\rangle = \sum_{k=0}^{N-1} \sqrt{\frac{|\alpha_k|}{\lambda}}\,|k\rangle\,U_k\,|\psi\rangle$$

- Aby poznać $|\psi_3\rangle$ przyjrzyjmy się jak działa $PREP^\dagger$:

$$\langle 0|\,PREP^\dagger = (PREP\,|0\rangle)^\dagger = \left(\sum_{k=0}^{N-1} \sqrt{\frac{|\alpha_k|}{\lambda}}\,|k\rangle\right)^\dagger = \sum_{k=0}^{N-1} \sqrt{\frac{|\alpha_k|}{\lambda}}\,\langle k|$$

Uzbrojeni w tą wiedzę dostajemy, że:

$$|\psi_3\rangle = \langle 0|\,PREP^\dagger \cdot |\psi_2\rangle = \left(\sum_{k=0}^{N-1} \sqrt{\frac{|\alpha_k|}{\lambda}}\,\langle k|\right) \cdot \left(\sum_{k=0}^{N-1} \sqrt{\frac{|\alpha_k|}{\lambda}}\,|k\rangle\,U_k\,|\psi\rangle\right)$$

Pogrupujmy pary wartości z obu sum względem tego czy są te same:

$$= \left(\sum_i \frac{|\alpha_i|}{\lambda} \langle i|i\rangle U_i |\psi\rangle \right) + \left(\sum_{i \neq j} \frac{\sqrt{|\alpha_i| \cdot |\alpha_j|}}{\lambda} \langle i|j\rangle U_j |\psi\rangle \right)$$

Zauważmy, że skoro $|i\rangle \perp |j\rangle$ to $\langle i|i\rangle = 1$ oraz $\langle i|j\rangle = 0$ dla $i \neq j$. Kasuje się prawy nawias i otrzymujemy:

$$= \sum_{i=0}^{N-1} \frac{|\alpha_i|}{\lambda} U_i |\psi\rangle = \frac{1}{\lambda} \sum_{i=0}^{N-1} \alpha_i U_i |\psi\rangle = \frac{A}{\lambda} |\psi\rangle$$

□

5 QFT

Kwantowa transformata Fouriera (ang. Quantum Fourier Transform) to przekształcenie analogicze do Dyskretnej Transformaty Fouriera (DFT), tylko w przypadku kwantowym. Przypomnijmy najpierw definicję DFT:

Definicja. (DFT)

Dyskretną transformatą Fouriera nazywamy funkcję

$$DFT : (x_0, \dots, x_{N-1}) \mapsto (y_0, \dots, y_{N-1})$$

gdzie

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j \omega^{-jk}$$

$$\omega = \sqrt[N]{1} = e^{2\pi i/N}$$

Podobnie możemy sformułować QFT.

Definicja. (QFT)

Niech $|x\rangle = \sum_{j=0}^{N-1} x_j |j\rangle$.
Kwantową transformatą Fouriera nazywamy funkcję

$$QFT : |x\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega^{xk} |k\rangle$$

Można jednak patrzeć na to w inny sposób. Niech macierz Fouriera F_N będzie określona następująco:

$$F_N = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \dots & \omega^{N-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \dots & \omega^{(N-1)(N-1)} \end{pmatrix}$$

Wtedy, przyjmując $X = (x_0, \dots, x_{N-1})$, $Y = (y_0, \dots, y_{N-1})$ mamy, że:

$$DFT(X) = X \cdot F_N = Y$$

Łatwo można sprawdzić, że macierz F_N jest unitarna. Zatem jest ona poprawną bramką kwantową. Oznacza to ni mniej ni więcej tyle, że QFT można traktować właśnie jako zaaplikowanie bramki F_N:

$$QFT |x\rangle \iff F_N |x\rangle$$

Teraz postaramy się skonstruować obwód dla F_N. Niech zapis $0.k_1k_2 \dots k_n$ oznacza liczbę zapisaną binarnie, przy pomocy bitów $k_1, \dots, k_n$. Wtedy zachodzi:

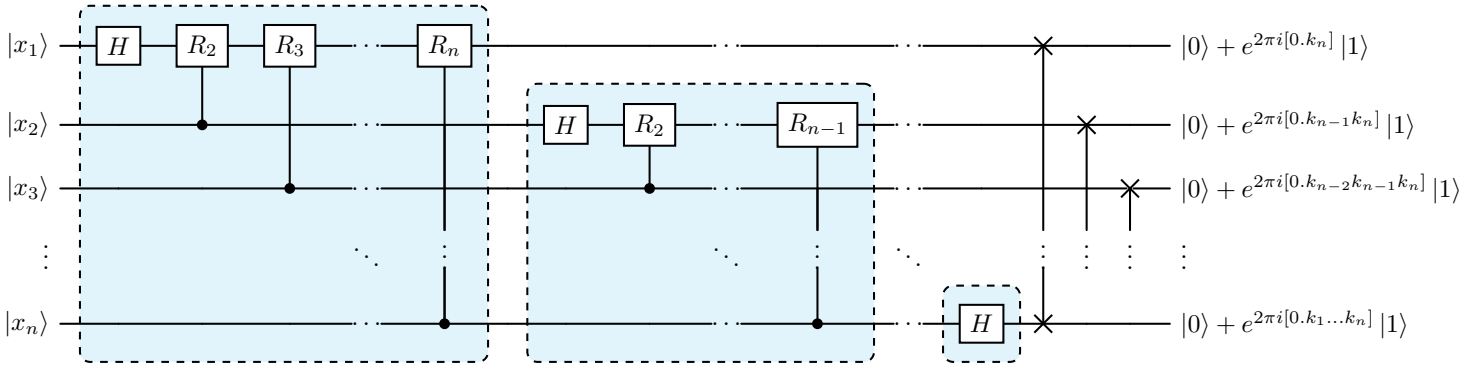
$$QFT |k\rangle = QFT |k_1\rangle \dots |k_n\rangle$$
$$= \frac{1}{\sqrt{N}} \left(|0\rangle + e^{2\pi i \cdot 0.k_n} |1\rangle\right) \oplus \left(|0\rangle + e^{2\pi i \cdot 0.k_{n-1}k_n} |1\rangle\right) \oplus \dots \oplus \left(|0\rangle + e^{2\pi i \cdot 0.k_1 \dots k_n} |1\rangle\right)$$

Reprezentacja ta jest o tyle pomocna, że przedstawiamy wyjście jako produkt tensorowy pojedynczych kubitów. Będą nam teraz potrzebne jedynie dwie bramki: H i R_k:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$R_k = \begin{pmatrix} 1 & 0 \\ 1 & e^{2\pi i/2^k} \end{pmatrix}$$

Wtedy nasz układ prezentuje się następująco (na wyjściach dla czytelności pominięty współczynnik $\frac{1}{\sqrt{2}}$):



(9)

Dowód. Zauważmy, że przy naszych oznaczeniach $H = \frac{1}{\sqrt{2}}R_1$. Zatem, jeśli prześledzimy pojedynczy kabel dojdziemy do wniosku, że domnażając wszystko do siebie otrzymamy *prawie* to czego potrzebujemy. Prawie, bo wszystko będzie w odwrotnej kolejności. Dlatego potrzebujemy jeszcze na końcu zamienić bity miejscami bramkami $SWAP$. □

Nasz układ posiada n bramek H, $(n-1)+\cdots+2+1 = \frac{n(n-1)}{2}$ R_k oraz $\lfloor \frac{n}{2} \rfloor SWAP$. Sumarycznie mamy zatem $\mathcal{O}(n^2)$ bramek.
Jest to znakomity wynik, gdyż w klasyczne DFT wymaga $\mathcal{O}(n2^n)$ operacji.

5.1 Quantum Phase Estimation

Problem

Dany jest unitarny operator U oraz stan własny $|\psi\rangle$. A zatem możemy go zapisać jako:

$$U|\psi\rangle = e^{2\pi i\phi}|\psi\rangle$$

Zadaniem jest wyznaczenie wartości ϕ.

Zauważmy, że skoro $\phi \in [0,1]$ to możemy traktować $|\phi\rangle$ jako stan odpowiadający zapisowi binarnemu ϕ. Zobaczmy co się stanie po zaaplikowaniu QFT na nim:

$$QFT|\phi\rangle = \frac{1}{\sqrt{2^n}}\sum_{k=0}^{2^n-1} e^{2\pi i\phi k}|k\rangle$$

Wygląda podobnie do naszego U, ale nie do końca. Zapamiętajmy jednak, że jak zobaczymy wyrażenie po prawej, to operacją $QFT^\dagger$ możemy odzyskać ϕ.
Mamy zatem superpozycję stanów $U^k|\phi\rangle$:

$$U^k|\psi\rangle = e^{2\pi i\phi k}|\psi\rangle$$

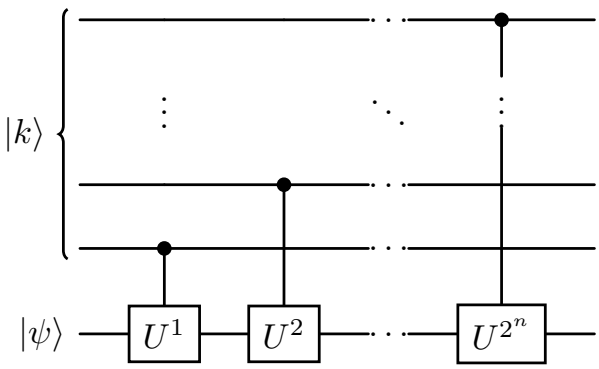
Zauważmy, że gdybyśmy znaleźli operator SEL' zachowujący się następująco:

$$SEL'|k\rangle|\psi\rangle = |k\rangle U^k|\psi\rangle$$

To umielibyśmy skonstruować cały układ:

$$|0\rangle_n|\psi\rangle \xmapsto{H^{\otimes n}\otimes I} \frac{1}{\sqrt{2^n}}\sum_{k=0}|k\rangle|\psi\rangle \xmapsto{SEL'} \frac{1}{\sqrt{2^n}}\sum_{k=0}|k\rangle U^k|\psi\rangle = \frac{1}{\sqrt{2^n}}\left(\sum_{k=0} e^{2\pi i\phi k}|k\rangle\right)|\psi\rangle \xmapsto{QFT^\dagger\otimes I} |\phi\rangle|\psi\rangle$$

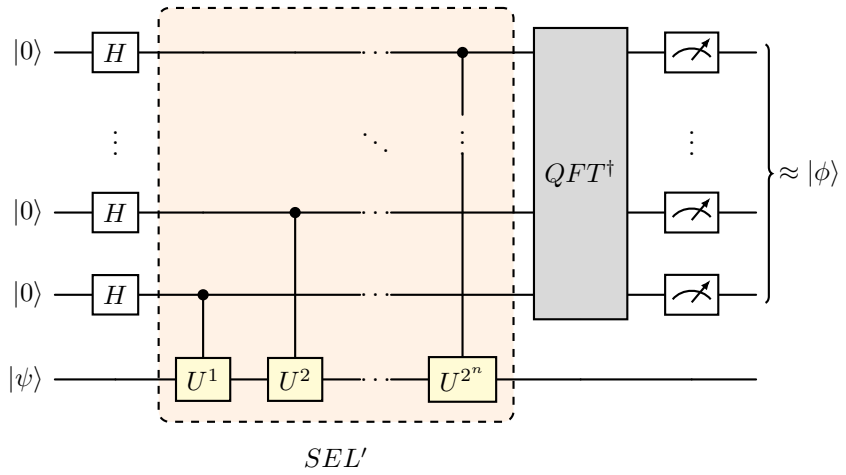
Okazuje się, że SEL' nie jest tak trudny do skonstruowania:



(10)

Jeśli przyjrzymy się ostatniemu kablowi i temu co do niego domnażamy to dojdziemy do wniosku, że faktycznie układ spełnia swoje zadanie. Stan $|k\rangle$ odpowiada zapisowi binarnemu k, a zatem jeśli odpowiedni bit i będzie zapalony w k to

domnożymy do wyniku odpowiednią potęgę U^{2^i}.
Mamy już wszystko. Możemy zatem podziwiać końcowy układ w pełnej okazałości:



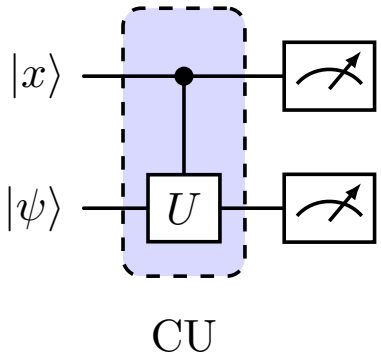
(11)

5.2 Phase kickback

Definicja

Phase kickback to sytuacja, w której kubity kontrolowane mają wpływ na kubity je kontrolujące.

Przykład. Rozważmy dowolną bramkę CU (controlled unitary):



(12)

Jest kontrolowana, czyli będzie się zachowywać następująco:

$$CU\,|0\rangle\,|\psi\rangle = |0\rangle\,|\psi\rangle$$
$$CU\,|1\rangle\,|\psi\rangle = |1\rangle\,U\,|\psi\rangle$$

Niech dodatkowo $|\psi\rangle$ będzie stanem własnym (eigenstate) U, czyli:

$$U\,|\psi\rangle = e^{2\pi i\phi}\,|\psi\rangle$$

dla pewnego ϕ. Ale uwaga, bo teraz jeśli popatrzymy dowolny stan $|x\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$ to otrzymujemy:

$$\begin{aligned} CU\,|x\rangle\,|\psi\rangle &= CU\,[(\alpha\,|0\rangle + \beta\,|1\rangle) \otimes |\psi\rangle] = \alpha CU(|0\rangle\,|\psi\rangle) + \beta CU(|1\rangle\,|\psi\rangle) \\ &= \alpha\,|0\rangle\,|\psi\rangle + \beta e^{2\pi i\phi}\,|1\rangle\,|\psi\rangle = \left(\alpha\,|0\rangle + \beta e^{2\pi i\phi}\,|1\rangle\right)|\psi\rangle \end{aligned}$$

Jeśli przypatrzymy się wejściu i wyjściu dojdziemy do ciekawego wniosku - "wynikowy" stan $|\psi\rangle$ ma wpływ na wartość pierwszego kubitu.
Właśnie tego typu efekt rozumiemy przez *phase kickback*.

5.3 Hamiltonian simulations

Jak wszyscy dobrze pamiętamy równanie Schrödingera wygląda z grubsza tak:

$$i\hbar\frac{d}{dt}\,|\psi(t)\rangle = H\,|\psi(t)\rangle$$

gdzie $|\phi_t\rangle = |\phi(t)\rangle$. Niech $|\psi_0\rangle$ oznacza stan początkowy.
Zakładając, że $H(t) = H$ to rozwiązanie przedstawia się następująco:

$$|\psi_t\rangle = e^{-itH/\hbar}\,|\psi_0\rangle = U(t)\,|\psi_0\rangle$$

Czyli:

$$U(t) = e^{-iHt}$$

Chcielibyśmy wyznaczyć $U(t)$. Jednak problemem jest rozmiar H oraz fakt, że chcemy policzyć $e^{H\cdots}$. Otóż H ma rozmiary $2^n \times 2^n$ co już dla dziesiątek kubitów daje olbrzymie macierze, z których jeszcze trzeba policzyć eksponens.
Z tego powodu wystarczy nam wyznaczenie $U'(t) \approx U(t)$. Aby pokonać problemy obliczeniowe skorzystamy z faktu, że:

$$e^{-iHt} = \lim_{L\to\infty} \left(e^{-i\frac{t}{L}H_1} e^{-i\frac{t}{L}H_2}\right)^L$$

gdzie $H = H_1 + H_2$. Będziemy próbowali tak ustalić H_1, H_2, by łatwo dało się z nich policzyć exp.
Na nasze potrzeby $L < \infty$, więc będą występować błędy. Niech $\Delta t = \frac{t}{L}$. Wtedy zachodzą następujące zależności:

$$||e^{-i\Delta tH} - e^{-i\Delta tH_1} e^{-i\Delta tH_2}|| = \mathcal{O}(\Delta t^2)$$

$$||e^{-itH} - e^{-itH_1} e^{-itH_2}|| = \mathcal{O}\left(\frac{t^2}{L}\right)$$

6 Algorytm Shor'a

6.1 Problem Simona

Problem

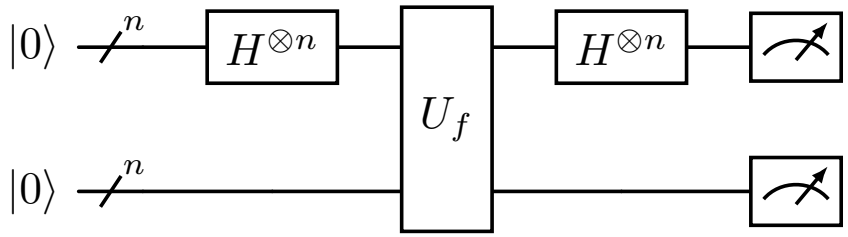
Dana jest czarna skrzynka w postaci funkcji $f : \{0,1\}^n \to \{0,1\}^m$, $m \geq n$. Mamy gwarancję, że istnieje wartość s, dla której:

$$f(x) = f(x') \iff x = x' \oplus s$$

Zadaniem jest wyznaczenie wartości s.
Chcemy również wykonać możliwie mało ewaluacji funkcji f.

Klasyczne podejście polegałoby na ciągłym ewaluowaniu f i obserwowaniu, czy nie otrzymaliśmy uzyskanej już wcześniej wartości. Jednak mamy aż 2^n argumentów do sprawdzenia. Nawet uwzględniając paradoks urodzin, rozwiązanie będzie wymagało średnio $\mathcal{O}\sqrt{2^n}$ ewaluacji funkcji f.

My jednak zaprezentujemy algorytm kwantowy, który pozwoli wyznaczyć s z dużym prawdopodobieństwem, przy użyciu $\mathcal{O}(n)$ zapytań o funkcję f. Głównym elementem będzie następujący obwód:



(13)

Sam algorytm przedstawia się następująco:

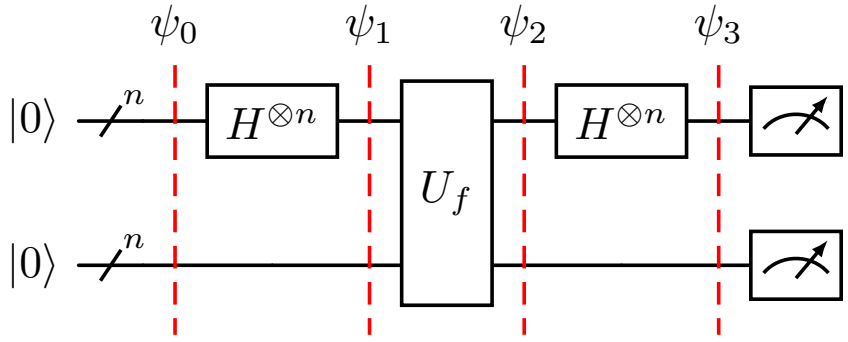
1. Uruchom powyższy układ odpowiednią ilość razy, generując zbiór liniowo niezależnych wektorów bitowych $y_1, \ldots, y_{n-1}$, **prostopadłych do** s.
2. Każde wygenerowane y_k spełnia $y_k \cdot s = 0$. Z tego układu równań wyznaczamy wartość s.

I to wszystko.
Pozostaje tylko kwestia, dlaczego to w ogóle działa. W tym celu pokażemy dwie rzeczy:

(i) Każdy wektor wygenerowany przez układ jest prostopadły do s.
(ii) Szansa, że wszystkie $y_1, \ldots, y_{n-1}$ są liniowo niezależne jest niemała.

Dowód. (i)

W tym celu musimy przeanalizować działanie układu krok po kroku:



(14)

- $|\psi_0\rangle = |0\rangle_n |0\rangle_n$
- Bramka Hadamarda działa tak:

$$H|0\rangle_n = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle$$

A więc:

$$|\psi_1\rangle = (H \otimes I_n)|\psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle |0\rangle_n$$

- Wyrocznia U_f działa tak:
$$U_f |x\rangle |y\rangle = |x\rangle |y \oplus f(x)\rangle$$
Zatem:
$$|\psi_2\rangle = U_f |\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle |f(k)\rangle$$
- Hadamard działa też tak:
$$H |k\rangle = \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} (-1)^{k\cdot j} |j\rangle$$
To daje nam
$$|\psi_3\rangle = (H \otimes I_n) |\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} \left[\frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} (-1)^{k\cdot j} |j\rangle \right] |f(k)\rangle$$
$$= \sum_{j=0}^{2^n-1} |j\rangle \left[\frac{1}{2^n} \sum_{k=0}^{2^n-1} (-1)^{j\cdot k} |f(k)\rangle \right]$$
A zatem prawdopodobieństwo zmierzenia danego stanu $|j\rangle$ jest równe:
$$\left\| \frac{1}{2^n} \sum_{k=0}^{2^n-1} (-1)^{j\cdot k} |f(k)\rangle \right\|^2$$

1) $s = 0$:
Wtedy mamy, że f jest bijekcją. Z tego mamy, że:
$$\left\| \frac{1}{2^n} \sum_{k=0}^{2^n-1} (-1)^{j\cdot k} |f(k)\rangle \right\|^2 = \frac{1}{4^n} \sum_{k=0}^{2^n-1} \left\| (-1)^{j\cdot k} |f(k)\rangle \right\|^2 = \frac{1}{4^n} \cdot 2^n = \frac{1}{2^n}$$

2) $s \neq 0$:
Wynika z tego, że $f(x_1) = f(x_2) = z$ dla pewnych x_1, x_2 oraz $z \in range(f)$. Czyli:
$$\left\| \frac{1}{2^n} \sum_{k=0}^{2^n-1} (-1)^{j\cdot k} |f(k)\rangle \right\|^2 = \left\| \frac{1}{2^n} \sum_{z\in range(f)} \left((-1)^{j\cdot x_1} + (-1)^{j\cdot x_2} \right) |z\rangle \right\|^2$$
Korzystamy z faktu, że $x_2 = x_1 \oplus s$:
$$= \left\| \frac{1}{2^n} \sum_{z\in range(f)} \left((-1)^{j\cdot x_1} + (-1)^{j\cdot (x_1 \oplus s)} \right) |z\rangle \right\|^2$$
$$= \left\| \frac{1}{2^n} \sum_{z\in range(f)} \left((-1)^{j\cdot x_1} + (-1)^{j\cdot x_1 \oplus j\cdot s} \right) |z\rangle \right\|^2$$
$$= \left\| \frac{1}{2^n} \sum_{z\in range(f)} (-1)^{j\cdot x_1} \left(1 + (-1)^{j\cdot s} \right) |z\rangle \right\|^2$$
W tej postaci jest to łatwe to przeanalizowania:
- Jeśli $j \cdot s = 1$, to wyrażenie ewaluuje się do 0.
- Jeśli $j \cdot s = 0$, to wyrażenie (postępując w analogiczny sposób jak w 1), pamiętając, że $|range(f)| = 2^{n-1}$) jest równe $\frac{1}{2^{n-1}}$.

Czyli możemy zmierzyć tylko takie $|j\rangle$, dla których zachodzi $j \cdot s = 0$. □

Dowód. (ii)
Dane są wektory $y_1, \ldots, y_{n-1}$. Jakie jest prawdopodobieństwo, że wszystkie są liniowo niezależne?
Wektorów prostopadłych do s jest 2^{n-1}. Wcześniej wyliczyliśmy też, że każdy z nich ma tą samą szansę bycia wylosowanym.
Zapytajmy się zatem dla każdego wektora y_k jaka jest szansa, że jest on liniowo niezależny z każdym z $y_1, \ldots, y_{k-1}$.

Dla y_1 jedyna sytuacja, że wektor będzie liniowo zależny to gdy wylosuje się $000\ldots0$. Szansa na to wynosi $\frac{1}{2^{n-1}}$, czyli szansa że jest lin. niez. wynosi $1 - \frac{1}{2^{n-1}}$
Dla y_2 zła sytuacja wystąpi wtedy, gdy wylosuje się 0 lub y_1. Czyli szansa że y_2 będzie liniowo niezależny wynosi $1 - \frac{1}{2^{n-2}}$.

Kontynuując rozumowanie mamy, że y_i jest liniowo niezależne od poprzednich z prawd. $1 - \frac{1}{2^{n-i}}$. Zatem szansa, że wszystkie wektory są niezależne wynosi:
$$\prod_{i=1}^{n-1} \left(1 - \frac{1}{2^{n-i}} \right) = \prod_{k=1}^{n-1} \left(1 - \frac{1}{2^k} \right) \geq \prod_{k=1}^{\infty} \left(1 - \frac{1}{2^k} \right) \approx 0.289\cdots \geq \frac{1}{4}$$

Czyli nasz proces wybierania losowania kolejnych wektorów kończy się sukcesem w co najmniej $\frac{1}{4}$ przypadków. Czyli oczekiwana liczba powtórzeń do uzyskania sukcesu wynosi co najwyżej $1/\frac{1}{4} = 4 = \mathcal{O}(1)$.

□

6.2 Algorytm Shora

Problem

Dana jest liczba n.
Zadaniem jest znalezienie nietrywialnego jej dzielnika d.

Sam algorytm składa się z dwóch części:

1. Klasycznej, sprowadzającej problem faktoryzacji do problemu znajdywania *rzędu*[1] w grupie multiplikatywnej.
2. Kwantowej, znajdującej ten rząd w zadowalającym czasie

Część klasyczna

W dalszej części będziemy zakładać, że N jest liczbą nieparzystą niebędącą potęgą liczby pierwszej.

Obserwacja. Aby sprawdzić czy N jest postaci p^k wystarczy przeiterować się po wszystkich sensownych k, tj. $k \leq \lg N$ (dla $k > \lg N$ mamy $\sqrt[k]{N} < 2$). Dla każdego takiego k znajdujemy $p' = \sqrt[k]{N}$ (np. wyszukiwaniem binarnym). Jeśli takowa liczba p' jest pierwsza (co sprawdzimy np. Millerem-Rabinem) to liczba N była postaci p^k dla $p \in \mathbb{P}$.

Wtedy algorytm Shora przedstawia się następująco:

1. Losujemy $a \in [2, N-1]$
2. Wyliczamy $gcd(a, N)$. Jeśli $gcd(a, N) > 1$ to zwróć $d := gcd(a, n)$
3. Wyznaczamy r takie, że $a^r \equiv 1 \mod N$
   Jeśli r jest nieparzyste, wróć do kroku 1.
4. Wyliczamy $d := gcd(N, a^{r/2} + 1)$
   Jeśli d jest trywialny, wróć do kroku 1.
5. Zwróć d

Okazuje się, że dla N niebędącego potęgą liczby pierwszej

Część kwantowa

Jedyną niewiadomą w algorytmie pozostaje kwestia jak szybko wyznaczać rząd danego elementu.
TODO

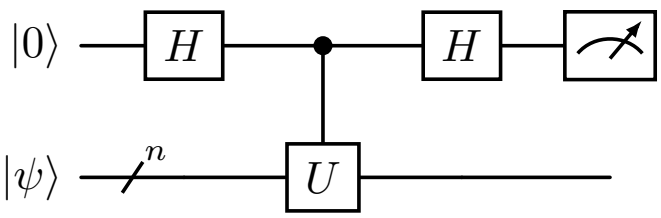
6.3 Test Hadamarda

Problem

Dany jest stan $|\psi\rangle$ oraz operator U. Naszym celem jest wyznaczenie $\operatorname{Re}\langle\psi| U |\psi\rangle$.

Motywacją jest fakt, że często będziemy chcieli poznać jaką *dokładnie* liczbę zespoloną produkuje iloczyn skalarny $\langle\phi|\psi\rangle$. Można oczywiście bezpośrednio mierzyć kubity na wyjściu, ale nie da nam to pełnej informacji, gdyż mierzymy jedynie amplitudę prawdopodobieństwa.

Test Hadamarda umożliwia nam stworzenie zmiennej losowej o wartości oczekiwanej równej $\operatorname{Re}\langle\psi| U |\psi\rangle$.
Układ prezentuje się następująco:



(15)

Jeśli zmierzymy na wyjściu $|0\rangle$ to wypluwamy 1. Jeśli zmierzymy $|1\rangle$ to wypluwamy -1.
Wtedy wartość oczekiwana na wyjściu jest równa dokładnie $\operatorname{Re}\langle\psi| U |\psi\rangle$.

Dowód. Przeanalizujmy co się kolejno dzieje w naszym układzie.

[1] **Rzędem** elementu a w grupie *mod* n nazywamy najmniejsze $r > 0$ takie, że $a^r \equiv 1 \ (mod\ n)$

Na wejściu mamy stan $|0\rangle|\psi\rangle$. Aplikujemy na górnym kablu Hadamarda, który działał tak:

$$H|0\rangle = \frac{1}{\sqrt{2}}\Big(|0\rangle + |1\rangle\Big), \qquad H|1\rangle = \frac{1}{\sqrt{2}}\Big(|0\rangle - |1\rangle\Big)$$

Czyli u nas:

$$|0\rangle|\psi\rangle \xmapsto{H\otimes I_n} \frac{1}{\sqrt{2}}\Big(|0\rangle + |1\rangle\Big)\otimes|\psi\rangle$$

Bramka U jest kontrolowana, czyli aplikuje się tylko dla $|1\rangle$ na kontrolującym kubicie:

$$\frac{1}{\sqrt{2}}\Big(|0\rangle + |1\rangle\Big)\otimes|\psi\rangle \xmapsto{CU} \frac{1}{\sqrt{2}}\Big(|0\rangle\otimes|\psi\rangle + |1\rangle\otimes U|\psi\rangle\Big)$$

Na końcu znowu aplikujemy Hadamarda:

$$\frac{1}{\sqrt{2}}\Big(|0\rangle\otimes|\psi\rangle + |1\rangle\otimes U|\psi\rangle\Big) \xmapsto{H\otimes I_n} \frac{1}{2}\Big((|0\rangle + |1\rangle)\otimes|\psi\rangle + (|0\rangle - |1\rangle)\otimes U|\psi\rangle\Big) =$$

$$= \frac{1}{2}\Big(|0\rangle\otimes(|\psi\rangle + U|\psi\rangle) + |1\rangle\otimes(|\psi\rangle - U|\psi\rangle)\Big)$$

$$= \frac{1}{2}\Big(|0\rangle\otimes(I+U)|\psi\rangle + |1\rangle\otimes(I-U)|\psi\rangle\Big)$$

Świetnie, zatem prawdopodobieństwo otrzymania $|0\rangle$ to kwadrat ze współczynnika przy nim swojącego. W ogólności jeśli stoi tam wektor stanów $|\phi\rangle$ to prawdopodobieństwo wynosi $\langle\phi|\phi\rangle = |\phi\rangle^\dagger|\phi\rangle$. Zatem liczymy prawdopodobieństwo p_0 zmierzenia $|0\rangle$:

$$p_0 = \frac{1}{4}\left((I+U)|\psi\rangle\right)^\dagger (I+U)|\psi\rangle = \frac{1}{4}\langle\psi|\left(I+U^\dagger\right)(I+U)|\psi\rangle$$

Jeśli powtórzymy analogiczne rozumowanie do policzenia prawdopodobieństwa p_1 zmierzenia na wyjściu $|1\rangle$ to otrzymamy, że:

$$p_1 = \frac{1}{4}\langle\psi|\left(I-U^\dagger\right)(I-U)|\psi\rangle$$

I teraz gwóźdź programu: traktujemy każde zmierzone $|0\rangle$ jako 1, a każde $|1\rangle$ jako -1. Oznaczmy tą zmienną losową przez X. Z definicji wartości oczekiwanej mamy:

$$E[X] = p_0\cdot(1) + p_1\cdot(-1)$$

$$= \frac{1}{4}\langle\psi|\left[\left(I+U^\dagger\right)(I+U) - \left(I-U^\dagger\right)(I-U)\right]|\psi\rangle$$

$$= \frac{1}{4}\langle\psi|\left[\cancel{I} + U + U^\dagger + \cancel{U^\dagger U} - \cancel{I} + U + U^\dagger - \cancel{U^\dagger U}\right]|\psi\rangle$$

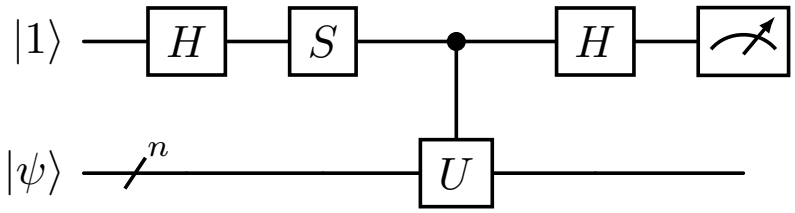
$$= \frac{1}{2}\langle\psi|\left(U+U^\dagger\right)|\psi\rangle \;=\; \frac{1}{2}\langle\psi|U|\psi\rangle + \frac{1}{2}\Big(\langle\psi|U|\psi\rangle\Big)^\dagger$$

Ale zauważmy, że $\langle\psi|U|\psi\rangle = z \in \mathbb{C}$. Zatem $\Big(\langle\psi|U|\psi\rangle\Big)^\dagger = z^\dagger = \bar{z}$. Ale z własności liczb zespolonych mamy, że $z + \bar{z} = 2\operatorname{Re} z$. Zatem:

$$E[X] = \frac{1}{2}(z+\bar{z}) = \operatorname{Re} z$$

□

Co jednak, jeśli chcielibyśmy zmierzyć wartość $\operatorname{Im}\langle\psi|U|\psi\rangle$? Okazuje się, że możemy tego dokonać w analogiczny sposób, nieznacznie modyfikując układ:



(16)

gdzie $S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$ - *Phase gate*. Działa ona w następujący sposób:

$$S|0\rangle = |0\rangle, \quad S|1\rangle = i|1\rangle$$

Otrzymujemy zatem:

$$|1\rangle|\psi\rangle \xmapsto{H\otimes I} \frac{1}{\sqrt{2}}\Big(|0\rangle - |1\rangle\Big)|\psi\rangle \xmapsto{S\otimes I} \frac{1}{\sqrt{2}}\Big(|0\rangle - i|1\rangle\Big)|\psi\rangle$$

Teraz postępujemy analogicznie jak wcześniej:

$$\xmapsto{CU} \frac{1}{\sqrt{2}}\Big(|0\rangle - i|1\rangle U\Big)|\psi\rangle$$

$$\xmapsto{H\otimes I} \cdots = \frac{1}{\sqrt{2}}\Big[|0\rangle\Big(|\psi\rangle - iU|\psi\rangle\Big) + |1\rangle\Big(|\psi\rangle + iU|\psi\rangle\Big)\Big]$$

Prawdopodobieństwo zmierzenia 0:

$$p_0 = \frac{1}{4}\Big(|\psi\rangle - iU\,|\psi\rangle\Big)^\dagger\Big(|\psi\rangle - iU\,|\psi\rangle\Big) = \frac{1}{4}\Big(\langle\psi| + iU^\dagger\,\langle\psi|\Big)\Big(|\psi\rangle - iU\,|\psi\rangle\Big)$$
$$= \frac{1}{4}\,\langle\psi|\,\Big(I + iU^\dagger\Big)\Big(I - iU\Big)\,|\psi\rangle$$

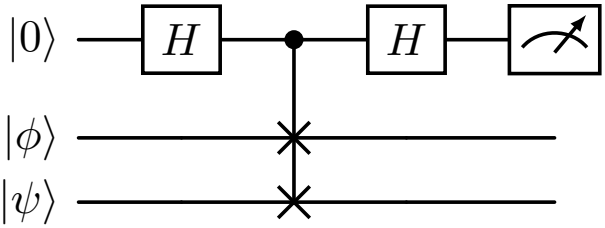
Powtarzając to samo rozumowanie jak dla standardowej wersji testu i pamiętając, że dla $z \in \mathbb{C}$ mamy $(-iz) + (\overline{-iz}) = 2\,\mathrm{Im}\,z$ dostaniemy, że układ faktycznie w oczekiwaniu wypluwa $\mathrm{Im}\,z$.

6.4 Test SWAP

Problem

Dane są dwa stany $|\phi\rangle$ oraz $|\psi\rangle$. Zadaniem naszym jest oszacowanie wartości $|\,\langle\phi|\psi\rangle\,|^2$.

Zauważmy, że test SWAP osiąga cel podobny do testu Hadamarda - pozwala nam ocenić, jak bardzo "podobne" są do siebie dwa stany $|\phi\rangle$ i $|\psi\rangle$.
Sam układ wygląda tak:



(17)

Główna bramka w nim występująca to kontrolowana bramka SWAP, która zamienia dwa dolne wejścia miejscami jeśli dostanie $|1\rangle$ na górnym kablu.
Jeśli każde zmierzone $|0\rangle$ potraktujemy jako 1, a każde $|1\rangle$ jako -1, to okaże się, że wartość oczekiwana takiej zmiennej losowej wynosi $|\,\langle\phi|\psi\rangle\,|^2$.

Dowód. Prześledźmy kolejne aplikacje bramek w naszym układzie:

$$|0\phi\psi\rangle \xmapsto{H\otimes I\otimes I} \frac{1}{\sqrt{2}}\Big(|0\phi\psi\rangle + |1\phi\psi\rangle\Big) \xmapsto{CSWAP} \frac{1}{\sqrt{2}}\Big(|0\phi\psi\rangle + |1\psi\phi\rangle\Big)$$
$$\xmapsto{X\otimes I\otimes I} \frac{1}{2}\Big(\big(|0\rangle + |1\rangle\big)\,|\phi\psi\rangle + \big(|0\rangle - |1\rangle\big)\,|\psi\phi\rangle\Big) = \frac{1}{2}\,|0\rangle\,\big(|\phi\psi\rangle + |\psi\phi\rangle\big) + \frac{1}{2}\,|1\rangle\,\big(|\phi\psi\rangle - |\psi\phi\rangle\big)$$

Liczymy prawdopodobieństwo p_0 uzyskania $|0\rangle$:

$$p_0 = \frac{1}{4}\Big(|\phi\psi\rangle + |\psi\phi\rangle\Big)^\dagger\Big(|\phi\psi\rangle + |\psi\phi\rangle\Big) = \frac{1}{4}\Big(\langle\psi\phi| + \langle\phi\psi|\Big)\Big(|\phi\psi\rangle + |\psi\phi\rangle\Big)$$
$$= \frac{1}{4}\Big(\langle\psi\phi|\phi\psi\rangle + \langle\psi\phi|\psi\phi\rangle + \langle\phi\psi|\phi\psi\rangle + \langle\phi\psi|\psi\phi\rangle\Big)$$
$$= \frac{1}{4}\Big(1 + |\langle\phi|\psi\rangle|^2 + |\langle\phi|\psi\rangle|^2 + 1\Big)$$
$$= \frac{1}{2} + \frac{1}{2}\,|\langle\phi|\psi\rangle|^2$$

gdzie $\langle\psi\phi|\psi\phi\rangle = \langle\phi\psi|\phi\psi\rangle = |\langle\phi|\psi\rangle|^2$ bierze się stąd:

$$\langle\psi\phi|\psi\phi\rangle = \langle\psi|\,\underbrace{\langle\phi|\psi\rangle}_{\in\,\mathbb{C}}\,|\phi\rangle = \langle\psi|\phi\rangle\,\langle\phi|\psi\rangle = \Big(\langle\phi|\psi\rangle\Big)^\dagger\,\langle\phi|\psi\rangle = |\langle\phi|\psi\rangle|^2$$

Z tego mamy, że prawdopodobieństwa p_1 otrzymania $|1\rangle$:

$$p_1 = \frac{1}{2} - \frac{1}{2}\,|\langle\phi|\psi\rangle|^2$$

Wprowadźmy teraz naszą zmienną losową X. Z prawdopodobieństwem p_0 otrzymujemy 1, a z prawd. p_1 dostajemy -1. Pozostaje wyznaczyć $E[X]$:

$$E[X] = p_0\cdot(1) + p_1\cdot(-1) = \cancel{\frac{1}{2}} + \frac{1}{2}\,|\langle\phi|\psi\rangle|^2 - \cancel{\frac{1}{2}} + \frac{1}{2}\,|\langle\phi|\psi\rangle|^2 = |\langle\phi|\psi\rangle|^2$$

□

Lemat

Dla ustalonej prezycji ε zarówno test Hadamarda jak i test SWAP wymagają $N = \mathcal{O}(1/\varepsilon^2)$ prób.

Dowód. Bierze się to z faktu, że oba N razy próbkują pewną zmienną losową, która zwraca jeden z możliwych wyników ze stałym prawdopodobieństwem p. Dla pojedynczej próby X_i wariancja $Var[X_i]$ jest stała, tj. $Var[X_i] = \mathcal{O}(1)$.

Każda próba jest niezależna od pozostałych, czyli dla $X = \frac{1}{N}\big(X_1 + \cdots + X_N\big)$ mamy:

$$Var[X] = Var\left[\frac{1}{N}\Big(X_1 + \cdots + X_N\Big)\right] = \frac{1}{N^2}Var[X_1 + \cdots + X_N] = \frac{1}{N^2}\sum_{i=1}^{N} Var[X_i] = \frac{1}{N^2}\mathcal{O}(N) = \mathcal{O}\left(\frac{1}{N}\right)$$

A odchylenie standardowe $\sigma(X) = \sqrt{Var[X]} = \mathcal{O}\left(\frac{1}{\sqrt{N}}\right)$. Czyli chcąc $\sigma(X) \leq \varepsilon$ potrzebujemy:

$$\sigma(X) \leq \varepsilon \iff \mathcal{O}\left(\frac{1}{\sqrt{N}}\right) \leq \varepsilon \iff \mathcal{O}\left(\frac{1}{N}\right) \leq \varepsilon^2 \iff N = \mathcal{O}\left(\frac{1}{\varepsilon^2}\right)$$

□

7 Stabilizatory

Definicja

Wyróżniamy cztery **bramki Pauliego**:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Zwane są też po prostu macierzami Pauliego. Mają parę ciekawych własności.

- Dwie różne bramki Pauliego, niebędące I, *antykomutują*, czyli:
$$XY = -YX, \;\; XZ = -ZX, \;\; YZ = -ZY$$
- Każda jest własną odwrotnością:
$$X^2 = Y^2 = Z^2 = I$$
- Ogólnie to z dokładnością do fazy tworzą zamkniętą grupę:
$$\begin{aligned} XY &= iZ & YX &= -iZ \\ YZ &= iX & ZY &= -iX \\ ZX &= iY & XZ &= -iY \end{aligned}$$
- Wszystkie są unitarne i hermitowskie.

Definicja

Grupą Pauliego dla pojedynczego kubitu nazywamy grupę

$$P_1 = \{\pm 1, \pm i\} \times \{X, Y, Z, I\}$$

Definicja działa również dla większej ilości kubitów:

$$P_n = \{\pm 1, \pm i\} \times \{X, Y, Z, I\}^{\otimes n}$$

Definicja

Mówimy, że operator unitarny U **stabilizuje** stan $|\psi\rangle$, jeśli $U|\psi\rangle = |\psi\rangle$.
Wtedy U nazywamy **stabilizatorem**.
Równoważnie można powiedzieć, że stan $|\psi\rangle$ jest **stabilizowany** przez operator U.

Definicja

Niech $|\psi\rangle$ to dowolny (określony) stan n-kubitowy.
Wtedy **grupa stabilizująca** G to podzbiór wszystkich P_n stabilizujących $|\psi\rangle$.

Mają one ciekawe własności. Załóżmy, że mamy grupę stabilizujacą $G \leq P_n$ na n kubitach. Wtedy:

- G jest abelowa ($PQ \in G \iff QP \in G$)
- $-I^{\otimes n} \notin G$

Definicja

Wymiar grupy G definiujemy następująco:

$$\dim(G) := \log_2 |G|$$

Równoważnie wymiar to ilość generatorów danej grupy.

W szczególności $\dim(G) = k \in \mathbb{N}$ jest równoważne temu, że $|G| = 2^k$.
Ale przecież jeśli $G \leq P_n$, to z tw. Lagrange'a mamy, że $|G|$ dzieli $|P_n|$. Ale (z definicji) $P_n = 4^{n+1}$. Oznacza to, że zawsze $G = 2^k$, dla pewnego $k \in \mathbb{N}$.

Definicja

Bramką Clifforda nazywamy bramkę kwantową U, dla której zachodzi:

$$UPU^{-1} \in P_n$$

dla każdego $P \in P_n$.

Przykłady.

- Bramka Hadamarda - ✓

Jest tak, gdyż:

$$HXH^\dagger = Z, \quad HYH^\dagger = -Y, \quad HZH^\dagger = X$$

A wszystkie wynikowe bramki oczwiście są Pauliego, czyli z definicji H jest bramką Clifforda.

- Bramka S - ✓

  Przypomnijmy, że $S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$. Wtedy:

$$SXS^\dagger = Y, \quad SYS^\dagger = -X, \quad SZS^\dagger = Z$$

- Bramka T - ✗

  Przypomnijmy, że $T = P(\pi/4) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}$. Ale mamy, że:

$$TXT^\dagger = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1+i}{\sqrt{2}} \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & \frac{1-i}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{1+i}{\sqrt{2}} & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & \frac{1-i}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0 & \frac{1-i}{\sqrt{2}} \\ \frac{1+i}{\sqrt{2}} & 0 \end{pmatrix}$$

  Ale nie jest to żadna z bramek Pauliego, więc T nie jest bramką Clifforda.

Fakt. (1) $Cliff_n = \langle H, S, CX \rangle$.

Gdzie zapis $\langle \dots \rangle$ oznacza grupę/zbiór generowane przez zadane bramki.

Fakt. (2) Zbiór bramek $Cliff + T$ jest *uniwersalny*

Twierdzenie. (Gotessman-Knill)

Obwody kwantowe złożone tylko z bramek Clifforda da się wielomianowo zasymulować na klasycznym komputerze.

8 Adiabatic QC

Na dzień dzisiejszy nie jesteśmy w stanie zaimplementować nawet prostych algorytmów kwantowych. A o Shorze czy Groverze to możemy zapomnieć. Problemem są błędy obliczeń, dekoherencja czy po prostu za mała liczba kubitów w systemach.
Nie wszystko jednak stracone, bo istnieją już *prawie* praktyczne algorytmy kwantowe służące do rozwiązywania zadań optymalizacyjnych. Działają one na trochę innej zasadzie niż te, z którymi się spotkaliśmy wcześniej. Z tego powodu wyróżnia się to inne podejście jako osobny paradygmat *adiabatycznych* obliczeń kwantowych (adiabatic QC).

Twierdzenie. (Variational Principle)

Niech H będzie znanym nam (niezależnym od czasu) Hamiltonianem. Niech $|\psi\rangle$ reprezentuje pewien stan kwantowy. Wtedy energię E_ψ stanu $|\psi\rangle$ wyliczamy następująco:

$$E_\psi = \langle\psi| H |\psi\rangle$$

Niech E_0 określa najmniejszą możliwą energię, osiąganą dla pewnego stanu bazowego.
Variational Principle mówi po prostu, że:

$$E_0 \leq \langle\psi| H |\psi\rangle$$

Definicja. (Barren plateaus)

Mianem **Barren plateaus** określamy zjawisko występowania nieproporcjonalnie dużych, (wykładniczo) płaskich regionów funkcji kosztu w kwantowych problemach optymalizacyjnych.

Fakt występowania tych równin jest o tyle niefortunny, że niepraktyczne staje się stosowanie tradycyjnego spadku wzdłuż gradientu. Niepraktyczne, bo gradient ten jest prawie zawsze równy 0, i algorytm nie jest w stanie znaleźć minimum. Duża część obecnych skupia się na tym by zrozumieć i przeciwdziałać fenomenowi równin Barrena.